Optimizing Automated Scoring for DAACS Writing Assessment with Large Language Modeling: Comparing Tokenizers and Models

Jason Bryer, Antonio Fraser, Julia Ferris, Angela Lui, and David Franklin

The Diagnostic Assessment and Achievement of College Skills (DAACS) is a suite of technological and social supports designed to optimize student learning. It consists of assessments in self-regulated learning (SRL), writing, mathematics, and reading whereby students receive immediate feedback in terms of one (developing), two (emerging), or three (mastering) dots which connects them to open educational resources (OER) to address any gaps in their college skills.

In the last few years there has been rapid development of natural language processing (NLP), especially with the availability of generative pre-trained transformer (GPT) models. The writing assessment is a key component of the DAACS intervention that serves two important purposes: 1) It provides students feedback in writing skills and strategies shown to be important for students to be successful in college; and 2) Encourages students to engage in self-regulatory processes since the assessment asks students to summarize their SRL results and identify strategies they will use to be successful. The current version of DAACS uses Lightside to score essays. This study explores more recent tokenizers and predictive modeling techniques to increase the accuracy of student scores as compared to human raters.

Machine scoring of essays is a two step process (Figure 1):

1. **Tokenization** - This is the process of converting the essay into a vector. The simplest tokenization procedure creates a column for each word present in all the training data where the value is the number times that word appears in that essay (Figure 2). Table 1 lists the tokenization procedures used in this study.
2. **Predictive Modeling** - This is the process of training a statistical model using the matrix generated in step one with the outcome variable, in this study the human scores. Table 2 lists the statistical models used in this study.

This study explores seven of the most common tokenizers with twelve different predictive modeling procedures.

Figure 1. Workflow for automated machine scoring

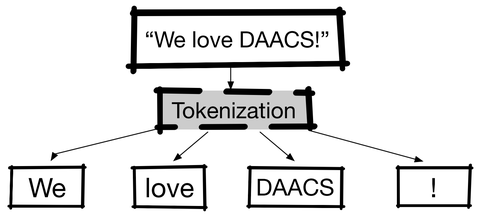

Figure 2. A tokenization illustration

Table 1. Tokenizers

Tokenizer Algorithm	Reference
BERT (Bidirectional Encoder Representations from Transformers)	Google (2018)
DistilBERT	Sanh et al., (2019)
Word 2 Vec	Google (2013)
Scikit Native Vectors	Pedregosa et. al (2012)
Facebook Vectors	Mojumder et al (2020)
Scikit Vectors	Bac et al., (2021)
Llama2 bytepair encoding	Meta (2023)

Table 2. Predictive models

Prediction Model	Reference
k-nearest neighbors	Fix & Hodges (1951)
Linear Support Vector (SVM)	Ratna et al (2019)
Classification	Dogra et al (2022)
C-Support Vector Classifiction	Novakovic et al (2011)
Gradient Boosting	Breiman (1997), Friedman (1999)
XGBoost	Shang, Men, and Du (2023)
Bagging	Islam et al (2022)
Random Forest	Breiman (2001)
AdaBoost	Sevinç (2022)
Logistic Regression	Cramer (2002)
Bayesian Classification (multinomial naive bayes from SK-learn)	Zang (2004)

Results

Figure 3 summarizes the accuracy for each tokenization and predictive modeling procedure. The top panel is organized by predictive modeling procedure where as the bottom panel is organized by tokenization procedure. This figure suggests that the predictive modeling choice may be more important than tokenization procedure. The distributions of accuracies within each tokenizer are very similar. From the top panel we can see that random forest and XGBoost, and to a lesser extent gradient boosting, perform well regardless of the tokenization procedure used.

Table 3 provides the baseline, human-to-human, accuracy used for training the predictive models and the best accuracy for Lightside used in version one of DAACS. The last set of columns are the tokenizer and predictive modeling procedures that produce the best overall accuracy. In seven of the nine domains the open source Llama developed by Facebook is the best performing tokenization procedure. In terms of predictive modeling, random forest is the best performing model in five of the nine domains.

Discussion

This study demonstrates the significant advances in machine scoring over the last several years. Although we don't believe machine scoring is ready for high stakes assessments, these results support the use of machine scoring for diagnostic purposes when the timeliness of feedback is critical.

Table 3. Accuracy for Human-to-human, Lightside, and natural language processing tokenizes

		LightSide		NLP		
Criteria	Human-Human	Model	Accuracy	Tokenizer	Model	Accuracy
Content Summary	55.8%	Logit	69.6%	Bert	Random Forest	71.7%
Conntent Suggestions	60.9%	Logit	72.3%	Llama	Gradient Boosting	72.5%
Sentences Structure	62.9%	Bayes	74.2%	Llama	XGBoost	78.5%
Sentences Transitions	57.1%	Bayes	47.2%	Bert	Random Forest	66.4%
Paragraphs Focus on a Main Ideas	59.8%	Logit	73.5%	Llama	Random Forest	75.0%
Paragraphs Cohesion	61.6%	Logit	72.7%	Llama	XGBoost	75.9%
Sentences Correct	56.3%	Logit	55.7%	Llama	XGBoost	70.9%
Sentences Complexity	56.0%	Bayes	68.4%	Llama	Random Forest	73.2%
Conventions	55.3%	Logit	63.2%	Llama	Random Forest	74.2%

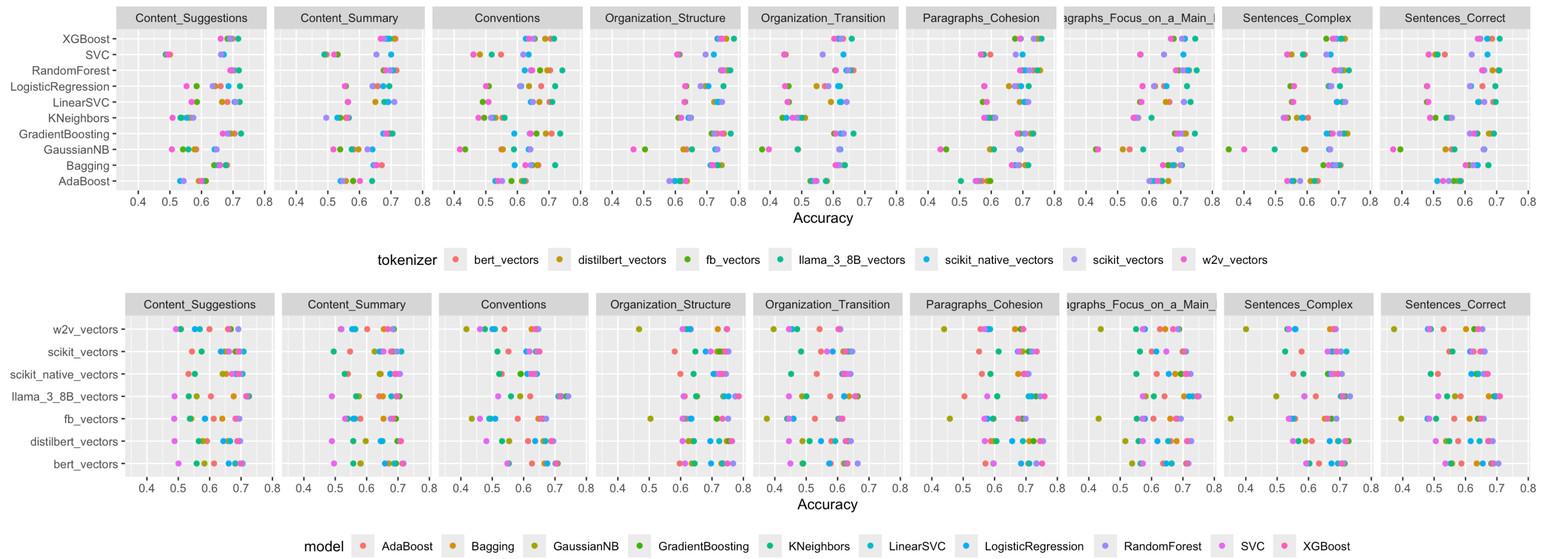

Figure 3. Scoring accuracy by tokenizer and predictive model.

jason.bryer@cuny.edu
https://github.com/DAACS/DAACS-NLP